

HOW IT WORKS

Lossy compression

Without compression our digital music, video and photo collections would overwhelm even the largest hard disks. **David Fearon** explains how compression loses the parts we'll never miss

Anyone who used computers in the 1990s will be familiar with compressed files. In those days, when disk space was scarce, ZIP files were an everyday way of cramming as much as possible on a floppy or hard disk.

ZIP, Stuffit and other file compression formats are known as lossless, because after decompression you get out exactly what you put in. These days, the most important and cutting-edge way of saving space is lossy compression, which doesn't give you exactly what you put in.

Lossy compression is the key to audio and video files that don't take an age to download and MP3 players that can store thousands of songs. Lossy compression leads to far smaller files.

ALTERED IMAGES

The main photo compression standard, JPEG, is named after the Joint Photographic Experts Group that developed it and, on the face of it, works like magic. An image from a 5-megapixel camera, saved as a lossless TIFF file, takes up 5-10MB of storage space. Convert that file to a JPEG and it shrinks to under 1MB with no apparent loss in quality.

The word 'apparent' is the key. JPEG files are this small because they use a 'perceptual encoding' scheme. When you view a JPEG image, you don't see a faithful copy of the original picture, but you don't notice. The system exploits the limitations of human perception to allow much of the information in the image to be thrown away.

The human eye is far more sensitive to brightness than colour, because we have more brightness-sensing rods in our eyes than colour-sensing cones. This means you can get away with discarding a large amount of the colour information in an image as long as you keep most of the brightness information.

This is done by taking the raw red, green and blue (RGB) values of each pixel as they're stored in a file, extracting their brightness and colour

information separately, then discarding much of the colour information. The image is also separated into tiled chunks, and frequency reduction is applied to remove details to which the eye is less sensitive (see below for more on frequency analysis). Finally, a process of lossless compression squeezes the remaining data down as much as possible.

MOTION BLUR

Video compression builds on the fundamentals of JPEG's lossy method. One way of compressing video is to treat each frame in isolation and apply a standard lossy image compression algorithm – typically JPEG – to each in turn. This is how the MJPEG and DV standards work. It compresses video moderately well but ignores the fact that most frames of a video are very similar to the one before and the one after; if they weren't, the video would be incomprehensible. A video of a newsreader, for instance, barely changes from one frame to the next: the lips move, and perhaps the head too, but most of the frame remains unchanged.

The most efficient way of compressing video is to record the first frame in its entirety and then for the second frame to record only the difference between it and the first. You can throw away most of the second image and then reconstruct during playback. The third frame can be made from the reconstructed second frame plus the difference, and so on. This method cuts out huge swathes of data.

When the scene changes completely, a full frame is inserted, known as a keyframe. In long sequences, particularly where lots of JPEG compression is used to make the keyframes as small as possible, the picture can gradually deteriorate as slight cumulative errors build up between frames. To solve this, keyframes are inserted at regular intervals before the video degrades noticeably. The number of keyframes depends on the encoder, but the standard DVD-quality keyframe sequence is every 12 or 15 frames,

while the maximum interval for DivX keyframe insertion is 300 frames.

There are many different video codecs, but all are based on the technique of lossily compressing the first frame and reducing the following frames by taking out static content. Many codecs, including QuickTime's MOV and the very popular DivX, are simply variations on the MPEG4 standard.

MPEG (named after the Moving Picture Experts Group) exploits the fact that video typically contains objects that persist in time. If you use vector maths to track the movement of objects through frames, a single frame can be used to reconstruct many frames. This technique is particularly effective at compressing camera pans. Reducing compression problems to vector calculations like this is great for making file sizes smaller, as vectors use far less data than bitmaps.

In MPEG compression, full keyframes are called I frames, and the 'difference' frames are called P frames. The system also uses backward-reference B frames, which enable the motion compensation to better predict the way things are going, improving prediction accuracy and hence image quality.

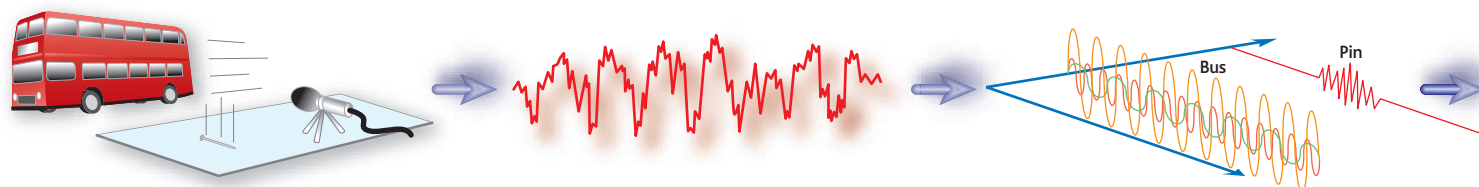
SOUNDS COMPLICATED

Audio is a different beast altogether. Unlike video, audio samples are not recorded in discrete frames that have meaning on their own. Rather than tending to stay the same, sound waves oscillate unpredictably. So while adaptive differential (AD) audio compression codecs exist, working like MPEG by guessing changes between samples, they're not very effective. A different method is needed.

The thing that tends to persist in sounds, and particularly music, is notes. Known technically as frequencies, notes have a duration in songs just as objects have a persistence in video. The trouble is that digitised audio doesn't actually record frequencies directly. Instead, it measures the sound pressure, or 'amplitude', which fluctuates wildly.

HOW FOURIER TRANSFORMS MAKE AUDIO COMPRESSION EASY

DIAGRAMS: MIKE HARDING



1 When sound is recorded everything from the roar of a bus to the sound of a pin drop is captured

2 In the sound recording all the noises are jumbled up. The wave oscillates in a seemingly random way

3 A Fourier Transform separates out the frequencies in the wave. Individual noises can now be discerned



FROM ANALOGUE TO DIGITAL: AUDIO SAMPLING EXPLAINED

To understand audio compression, it's important to understand the nature of audio sampling. Like most things in the real world, sound is analogue; it's a continuous wave. To record it digitally we need to break it into chunks that can be represented by numbers.

This is known as sampling. You take your continuous waveform and sample its height a number of times per second. If you do this quickly enough and use enough digital bits per sample to record the value accurately, you can capture a waveform so accurately that a human will find it hard to detect any loss in quality when it's played back as an analogue waveform.

The process of turning a continuous analogue value into a digital value is known as quantisation. You can increase the fidelity either by increasing the number of samples you take per second (the sampling frequency) or the accuracy of those samples (the sampling resolution). For instance, if you hold each sample value as one byte of data, you can record with very limited accuracy: one byte (eight digital bits) can represent only 256 values. When you play back the sound it will be raspy and thin with bizarre high-frequency overtones.

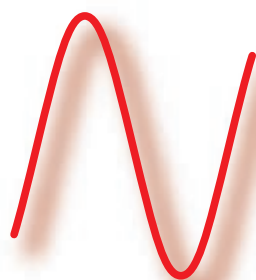
This is the negative effect of limited-resolution quantisation, known as aliasing distortion. If you use two bytes (16 bits) per sample, the sound quality improves as you have 65,536 values to play with and the chances are

the values you record digitally will be close to the source's analogue values in the real world. This is why standard CD audio is recorded at 16 bits per sample. CDs have a sampling frequency of 44.1kHz, or 44,100 samples per second, which is down to the Nyquist limit, a law of analogue to digital conversion that states that you can accurately digitise an analogue waveform if its maximum frequency is roughly half your sample

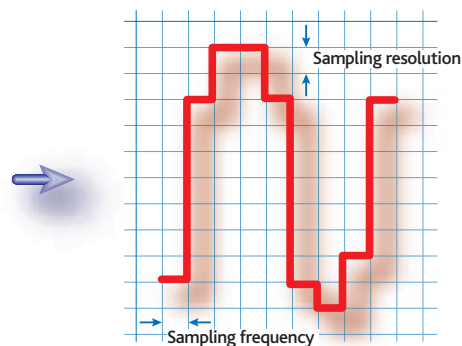
rate. As the limit of human hearing is around 20-22kHz, 44.1kHz is an ideal sample rate.

Nonetheless, some people claim that using a higher-resolution sample value and sample rate enhances sound quality. The audio chipsets on practically all new motherboards and sound cards support at least 96kHz/24-bit audio, and Intel's HD Audio standard can extend up to a ludicrous 192kHz/32-bit.

HOW SOUND WAVES ARE DIGITISED



1 Sound is a wave of air pressure. It's analogue, meaning the pressure varies continuously



2 A basic digital recording samples the pressure at fixed intervals and with a fixed level of accuracy

To find out which frequencies are present you need to process the signal with what's known as a frequency transform function. This splits the digital signal into its component frequencies while retaining information on the amplitude and duration of tones, allowing us to apply effective compression. The most popular and well-known function that does this is the Fourier transform.

Modern audio codecs are based on frequency analysis and what's known as psychoacoustic compression. This is a lossy compression scheme based on the same sort of principle as JPEG image compression. It takes advantage of limitations in human perception by discarding the data that most ears can't hear.

The most obvious place to start is with soft sounds that are masked by louder sounds. No-one can hear a pin drop when a bus is passing, but as those sounds occur in different parts of the

frequency spectrum both can be recorded simultaneously. Fourier analysis of the recorded audio signal reveals the pin dropping in a different part of the spectrum from the bus passing (see below). You can remove the data that relates to the pin as humans can't hear it. A human listener hears just the bus going past, but the perceptually useless data relating to the pin has been removed, reducing the size of the audio file.

However, there are also other types of perceptual masking. Temporal masking exists, too – a quiet sound that occurs immediately before or after a loud one will not be perceived by a human. Although you may think you perceive sounds at the precise moment they happen, you don't. Your brain has to process audio data, just as a computer does, and that takes time. If a loud sound occurs close to a quiet one, the brain alters its internal volume control and the quiet sound is swamped by

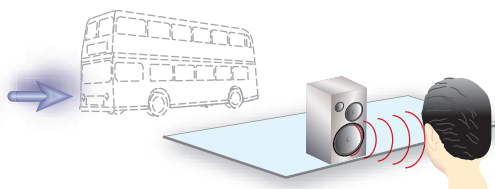
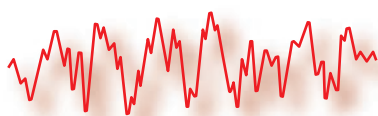
the loud one in that small chunk of processing time – up to about 50ms (fifty thousandths of a second).

CODEC MOMENTS

The many different popular audio codecs – MP3, WMA, Ogg Vorbis, Apple's AAC and Sony's ATRAC – are all variations on the theme of psychoacoustic lossy compression. They differ in their complexity and secondary features, such as digital rights management, but they're all fundamentally similar.

The older formats tend to sound worse because a key limitation for sophisticated compression is the computing power on hand to reconstruct the data. MP3 is the oldest format (ATRAC has been around longer but has been constantly revised) and needs the least computing power but arguably produces the worst sound quality. Microsoft's WMA has been constantly revised and sounds better than MP3 at lower bit rates, but you may notice that your audio player's batteries don't last as long when playing WMA; this is because the audio processor has to work harder on the more sophisticated WMA encoding.

You can achieve even better compression with variable bit-rate encoding, which is an option with most codecs. This simply lowers the number of samples per second in the moments when there are fewer high frequencies in the audio. Nyquist's theorem (see the box above) says that you need only 44.1kHz (or 44,100 samples per second) to represent frequencies up to 22kHz, so if at a given instant in the music the highest frequency present is, say, 10kHz, you can drop your sample rate by about half and so double the compression ratio for that short period. **CS**



4 You can remove noises a human listener wouldn't hear, then rebuild a smaller, simplified sound file

5 The sound file now only contains the bus noise. It's smaller, but sounds just like the original recording